scientific reports

OPEN

Sparse inference and active learning of stochastic differential equations from data

Yunfei Huang[1,4], Youssef Mabrouk[2,3,4], Gerhard Gompper[1] & Benedikt Sabass[1,2]✉

Automatic machine learning of empirical models from experimental data has recently become possible as a result of increased availability of computational power and dedicated algorithms. Despite the successes of non-parametric inference and neural-network-based inference for empirical modelling, a physical interpretation of the results often remains challenging. Here, we focus on direct inference of governing differential equations from data, which can be formulated as a linear inverse problem. A Bayesian framework with a Laplacian prior distribution is employed for finding sparse solutions efficiently. The superior accuracy and robustness of the method is demonstrated for various cases, including ordinary, partial, and stochastic differential equations. Furthermore, we develop an active learning procedure for the automated discovery of stochastic differential equations. In this procedure, learning of the unknown dynamical equations is coupled to the application of perturbations to the measured system in a feedback loop. We show that active learning can significantly improve the inference of global models for systems with multiple energetic minima.

Throughout the natural sciences, mathematical models are frequently formulated as differential equations. For example, with stochastic, ordinary, and partial differential equations (SDEs, ODEs, and PDEs). In physics, governing differential equations are often derived from first principles, for instance, from conservation of energy and momentum, and thermodynamic considerations. However, for complex systems studied, e.g., in biophysics, climate science, and neuroscience, first principles determining the system properties are typically not fully known. For example, because such systems are in a driven non-equilibrium state, highly nonlinear, and because dynamics may occur on multiple scales that are not well separated. In these cases, one can resort to phenomenological, effective descriptions that may result from some level of coarse graining and are based on experimental data. Recently, the increased availability of computational power has made it possible to construct such models in an automated fashion, which is known as data-driven discovery of governing equations.

Various approaches have been developed for inferring the differential equations that govern a non-linear dynamical system directly from measured data[1–6]. In a popular approach that is called "symbolic regression", function libraries are employed to automatically extract the terms in a governing equation that best represents the measured data according to some optimization criterion[1,2]. Recently, the use of sparse regression techniques for symbolic regression has received considerable scientific attention[3,4]. In symbolic regression, the physical quantity z, which is for illustration taken to be a scalar here, is assumed to obey an equation of the general form

$$\check{z} = \mathcal{F}(z, \mathbf{x}, t, c), \tag{1}$$

where $\check{z}$ can be, e.g., a time derivative $\check{z} = \frac{\partial z}{\partial t}$ for describing an ODE. $\mathcal{F}(z, \mathbf{x}, t, c)$ is an unknown function whose arguments $\mathbf{x}$ represent space coordinates while t represents time and c is a constant parameter. Vectors and arrays are denoted by bold letters. The aim of symbolic regression is to estimate the function $\mathcal{F}(\ldots)$ from a data set $\mathbf{z}$, which could be a measured sequence of values of z at different time-space coordinates. The vector $\check{\mathbf{z}}$ is either measured or estimated from $\mathbf{z}$, e.g., with a discrete difference scheme. For inference of $\mathcal{F}(\cdots)$, a so-called "library" matrix $\mathbf{\Theta}(z)$ is constructed from a suitable set of functions of $\mathbf{z}$, e.g., various powers of $\mathbf{z}$, combinations of partial derivatives, or trigonometric functions. Assuming that the governing Eq. (1) can be expressed as a linear superposition of library terms, we write

[1]Theoretical Physics of Living Matter, Institute of Biological Information Processing and Institute for Advanced Simulation, Forschungszentrum Juelich, 52425 Juelich, Germany. [2]Department of Veterinary Sciences, Institute for Infectious Diseases and Zoonoses, Ludwig-Maximilians-Universitaet Munich, 80539 Munich, Germany. [3]Helmholtz Institute Muenster (HI MS), IEK-12 Forschungszentrum Juelich GmbH, Corrensstraße 46, 48149 Muenster, Germany. [4]These authors contributed equally: Yunfei Huang and Youssef Mabrouk. ✉email: b.sabass@fz-juelich.de

$$\check{\mathbf{z}} = \mathbf{\Theta}(z)\boldsymbol{\xi}, \tag{2}$$

where $\boldsymbol{\xi}$ is a weight vector. The inference of the governing equation is thus reduced to a regression problem for the optimal $\boldsymbol{\xi}$, given $\check{\mathbf{z}}$ and $\mathbf{\Theta}(z)$. In general, solving the inverse problem in Eq. (2) is not straight-forward since the matrix $\mathbf{\Theta}$ should represent many equation terms and can have a large condition number $\kappa(\mathbf{\Theta})$.

In Ref.[3], a method called sparse identification of nonlinear dynamics (SINDy) has been proposed. The method works iteratively. At each iteration, $\boldsymbol{\xi}$ is first obtained from a least-squares optimization involving Eq. (2) and $\boldsymbol{\xi}$ is subsequently thresholded such that values smaller than a cutoff $\varkappa$ are set to zero. The iteration is continued until convergence conditions are satisfied. SINDy has been shown to be a powerful and versatile method that is applicable for inference of various types of ODEs[3]. However, the method requires the user to manually select the thresholds $\varkappa$. For the identification of PDEs, an alternative algorithm called train sequential threshold ridge regression (TrainSTRidge) has been described in Ref.[4]. This method is a variant of a least-squares optimization procedure for ridge regression called Sequential Threshold Ridge regression (STRidge). In STRidge, the vector $\boldsymbol{\xi}$ is first calculated by using ridge regression with a fixed regularization parameter. Then, all elements in $\boldsymbol{\xi}$ that have a smaller absolute value than a threshold $\varkappa$ are set to zero. Both, the regularization parameter and the threshold $\varkappa$ need to be provided by the user in STRidge. TrainSTRidge[4] employs L0 regularization and a training step to automatically determine the threshold $\varkappa$ while the regularization parameter remains to be set by the expert user. Conversely, a method called threshold sparse Bayesian regression, which also was employed for identification of PDEs[7], requires no input of a regularization parameters but some thresholds remain to be provided by the user.

The first aim of this work is to provide a method to solve the inverse problem associated with data-driven discovery of governing physical equations, Eq. (2), by combining a Bayesian approach with a automatic threshold-ing procedure. We call this method *automatic threshold sparse Bayesian learning* (ATSBL). Our algorithm does not require any manual fine-tuning of parameters to correctly infer governing differential equations from measured data. The method can be employed to identify ODEs, PDEs, and SDEs.

The case of SDEs requires particular attention, since the above-mentioned methods of equation inference are mainly designed for deterministic processes and some moderate amount of additive noise. The question of how to reconstruct the force fields for stochastic processes has been investigated in numerous studies, e.g., for application in soft matter physics and biophysics[5,8–13]. Recently, sophisticated methods have been proposed for dealing with discretization and the inference problem in the context of SDEs for second-order dynamics[14–17]. Here, we focus on the use of symbolic regression for the inference of analytical expressions of SDEs of the overdamped Langevin-type. One approach to symbolic regression in this context is based on dividing the phase space into small hypercubes which are also called bins in the one-dimensional case. Average values of the state variables and of their derivatives are estimated in each hypercube and the regression is defined with respect to these averages[5]. This kind of averaging generally depends on the chosen discretization and the averaging may lead to a substantial loss of information. Furthermore, application of this method to non-stationary processes requires a large ensemble of trajectories and considerable numerical effort to sample the time-dependent prob-ability distribution in phase space. The difficulties related to the averaging in phase space motivate the investiga-tion of the question to what extent the above-mentioned inference tools can be used in the context of noisy data without the need to perform ad hoc averaging, and, eventually, how the robustness of the inference methods may be improved in this context. We show that imposing Laplacian or Gaussian prior distributions on the inferred models is generally sufficient to identify the correct SDEs directly from trajectories without phase-space bin-ning and we provide a comparison of the accuracy of results obtained with the two types of prior distributions. A remarkable performance of the Laplacian prior is demonstrated with several examples, including Brownian motion in time-dependent potentials.

A major challenge for the inference of SDEs is that the phase space is often sampled very inhomogeneously in available data. This problem is encountered, e.g., for systems where the long-term dynamics is dominated by transitions between different, locally stable states, while the short-term dynamics are dominated by fluctuations around individual stable states. In such cases, the inferred equation may be meaningful only locally, i.e, within the region covered by the measurement trajectory, and it may be a priori impossible to infer the global dynamics from a given data set. To enable an automatic inference of a global model under these conditions, we consider the question of how to design an external perturbation to the system, also called "control force", such that the state variables are forced to explore the full phase space in a shortened sampling time. Established Umbrella sampling routines used for this purpose rely on quadratic control forces and involve non-trivial design steps for the control force[18–21]. See, e.g., Refs.[22,23] for alternative approaches. This kind of methodology has proved useful, e.g., in the context of computational studies of nucleation[24] and growth[25] processes. We develop an alternative adaptive control technique that recursively infers the governing equation and adapts the external control solely based on inferred equations. The adaptation loop consists of inference of the governing equation and a subse-quent update of the control force such that it is directly opposite to the inferred force. No parameters need to be tuned for designing the control with this adaptive scheme. Using the adaptive control scheme, we demonstrate a substantial improvement of the inference of SDEs for several different simulations of Brownian motion.

This work is organized as follows. The "Methods" section provides details on the the construction of function libaries and the casting of the inference problem into a system of linear equations. The inference algorithm is summarized and it is explained how Laplacian prior distributions can be used to impose the sparsity condition on the inferred models. In the "Results" section, the performance of the described method is illustrated by means of numerical examples and a comparison with previously described methods is presented. An adaptive sampling technique for improving the inference of SDEs is proposed and the usefulness of this approach is demonstrated.

Methods for data-driven identification of differential equations

Ordinary and partial differential equations.
Measurement data from a system of interest is presumed to be recorded as a time series of states, for example, a time-dependent position vector. In a data-driven approach to model a system, the data is used to automatically infer the a priori unknown dynamical equations that govern the observed process. In this work, inference is based on libraries of candidate functions for the governing equations. The data used for inference of differential equations is assumed to contain additive noise but no systematic errors.

For inference of ODEs, we generalize the introductory example for a scalar variable z, Eq. (1), to a system with M compontents that are assumed to be sampled with the same regular time interval for all $\ell \in \{1 \ldots M\}$ components. To distinguish discrete measurements from continuous variables, a subscript notation is employed in the following. The ℓ-th component measured in an ordered time series $[t_1, \ldots, t_N]$ is written as $\mathbf{z}_\ell = [z_{\ell,t_1}, z_{\ell,t_2}, \ldots, z_{\ell,t_N}]$. Vectors or arrays containing multiple variable measurements, e.g., at different time points, are denoted with bold letters. The whole data can then be written in matrix form as

$$\mathbf{Z} = [\mathbf{z}_1^{\mathrm{T}}, \ \mathbf{z}_2^{\mathrm{T}}, \ \ldots, \ \mathbf{z}_M^{\mathrm{T}}] = \begin{array}{c} \text{state} \end{array} \begin{bmatrix} z_{1,t_1} & z_{1,t_2} & \cdots & z_{1,t_N} \\ z_{2,t_1} & z_{2,t_2} & \cdots & z_{2,t_N} \\ \vdots & \vdots & \ddots & \vdots \\ z_{M,t_1} & z_{M,t_2} & \cdots & z_{M,t_N} \end{bmatrix}.$$

Our approach also requires derivatives of the measured data. For simplicity, finite-difference approximations are used throughout this work. Approximate derivatives are denoted by the operator $\mathcal{D}_{...}$, which represents here a fourth-order finite central difference scheme. For example, a time derivative of the ℓ-th state component, $\mathbf{z}_\ell$, at the i-th timepoint t_i is written as $\dot{z}_\ell(t)|_{t=t_i} \approx \mathcal{D}_t \mathbf{z}_\ell|_{t=t_i}$. For the entire dataset, we write the time derivative as

$$\dot{\mathbf{Z}} \approx \mathcal{D}_t \mathbf{Z} = \left[\mathcal{D}_t \mathbf{z}_1^{\mathrm{T}}, \ \mathcal{D}_t \mathbf{z}_2^{\mathrm{T}}, \ \ldots, \ \mathcal{D}_t \mathbf{z}_M^{\mathrm{T}} \right].$$

A governing ODE for the vector containing the trajectory of the ℓ-th state component may be written as a linear combination of elementary functions of all $\{\mathbf{z}_{\ell'}\}$, e.g., as

$$\mathcal{D}_t \mathbf{z}_\ell = \mathcal{F}_\ell(\{\mathbf{z}_{\ell'}\}, \ \{\mathbf{z}_{\ell''} \odot \mathbf{z}_{\ell'''}\}, \ \ldots, \ \{\cos \mathbf{z}_{\ell'}\}, \ \ldots, c), \tag{3}$$

where the indices ℓ', ℓ'', and ℓ''' cover the M system dimensions, $\odot$ denotes an element-wise product, and c represents a constant. $\mathcal{F}_\ell$ can also depend on time, but we focus mostly on autonomous differential equations in the following. Since $\mathcal{F}_\ell(\cdot)$ represents a linear combination of functions that can be calculated from the data, $\mathcal{F}_\ell(\cdot)$ can be expressed with the help of a library matrix $\boldsymbol{\Theta}(\mathbf{Z})$ multiplied with a sparse vector $\boldsymbol{\xi}_\ell$. Thus, we obtain for Eq. (3) in discretized form

$$\mathcal{D}_t \mathbf{z}_\ell = \boldsymbol{\Theta}(\mathbf{Z}) \boldsymbol{\xi}_\ell, \tag{4}$$

where the terms of the library matrix $\boldsymbol{\Theta}(\mathbf{Z})$ are calculated from the measurement data by evaluating the functions of $\{\mathbf{z}_{\ell'}\}$ and the non-zero elements of $\boldsymbol{\xi}_\ell$ characterize the dynamics of the system. Since Eq. (4) refers to ODEs, no derivative terms are contained in the library on the right-hand side of the equation. Given $\mathcal{D}_t \mathbf{z}_\ell$ and $\boldsymbol{\Theta}(\mathbf{z})$, the aim is to calculate a sparse vector $\boldsymbol{\xi}_\ell$ with a minimal number of non-zero coefficients corresponding to a minimal number of terms necessary to describe the dynamics.

For inference of PDEs, the library matrix $\boldsymbol{\Theta}$ has to contain partial-derivative terms. Thus, data is required that allows the numerical estimation of derivative expressions with respect to two or more variables, for example, with respect to time and space. Usually, measurements therefore consist of discrete space-time series recordings of system variables. For example, an array $\mathbf{Z}^P$ representing the M-dimensional state vector that is measured at N time points in R positions of one space coordinate x is written as

$$\mathbf{Z}^P = \begin{array}{c} \text{state} \end{array} \begin{bmatrix} z_{1,(t_1,x_1)} & z_{2,(t_1,x_1)} & \cdots & z_{N,(t_1,x_1)} \\ \vdots & & & \\ \end{bmatrix}$$

With a finite-difference approximation, vectors of time derivatives of every component, $\mathcal{D}_t \mathbf{z}_\ell$, and various orders of x derivatives are calculated, for example, $\mathcal{D}_x \mathbf{Z}^P$, $\mathcal{D}_{xx} \mathbf{Z}^P$, $\ldots$. These derivative terms are added to the

3

library $\boldsymbol{\Theta}^P$. Like for ODEs, inference of the dynamical equation governing the component $\mathbf{z}_\ell$ is then based on the linear equation

$$\mathcal{D}_t \mathbf{z}_\ell = \boldsymbol{\Theta}^P(\mathbf{Z}^P, \mathcal{D}_x \mathbf{Z}^P, \mathcal{D}_{xx} \mathbf{Z}^P, \ldots) \boldsymbol{\xi}_\ell, \tag{5}$$

with a sparse coefficient vector $\boldsymbol{\xi}_\ell$ to be determined.

Note that a robust estimation of derivatives from noisy data is an important prerequisite for data-driven inference of ODEs and PDEs in this framework. The fourth order finite-difference approximations employed here may be supplemented or replaced with other methods, including denoising procedures and Gaussian-process regression models.

Stochastic differential equations. We focus on Langevin-type SDEs to describe the time evolution of continuous, real state variables $\mathbf{X}(t)$, representing, e.g., the position of a Brownian particle in space[26]. Trajectories, denoted by $\mathbf{X}(t)$, are time-ordered sequences of values of space coordinates $\mathbf{x}$. The general form of the considered SDEs is

$$dX_\ell(t) = \underbrace{g_\ell(\mathbf{X}(t), t) dt}_{\text{deterministic part}} + \underbrace{h_{\ell,\ell'}(\mathbf{X}(t), t) \, dW_{\ell'}(t)}_{\text{noise}}, \tag{6}$$

where we employ the Einstein sum convention and $X_\ell(t)$ denotes the ℓ-th component of the system state at time t. The trajectories $\mathbf{X}$ are calculated by making use of Ito's interpretation of stochastic integrals[26]. The $g_\ell(\mathbf{X}(t), t)$ represent the deterministic parts of the differential equations. For example, for a Brownian particle undergoing overdamped motion in the presence of conservative forces with a potential $U(\mathbf{x}, t)$, we have $g_\ell(\mathbf{X}, t) = -\nabla_{x_\ell} U(\mathbf{x}, t)|_{\mathbf{x} = \mathbf{X}(t)}$. The stochastic perturbations are assumed to result from a Wiener process with a noise source $\Gamma_\ell(t)$ and $dW_\ell = \Gamma_\ell(t) \, dt$. The noise is assumed to obey a Gaussian distribution with a vanishing mean and a δ-correlated variance as

$$\langle \Gamma_\ell(t) \rangle = 0, \tag{7a}$$

$$\langle \Gamma_\ell(t) \Gamma_{\ell'}(t') \rangle = \delta_{\ell,\ell'} \delta(t - t'), \tag{7b}$$

respectively. The coefficient matrix $h_{\ell,\ell'}$ in Eq. (6) scales the magnitude of the stochastic perturbations and is assumed to be diagonal, for simplicity. Further noise sources, e.g., resulting from an experimental measurement of a trajectory, are not explicitly considered throughout this work.

The Fokker–Planck equation that corresponds to Eq. (6) and describes the evolution of a probability density function $f(\mathbf{x}, t)$ is given by

$$\frac{\partial f(\mathbf{x}, t)}{\partial t} = \hat{L} f(\mathbf{x}, t), \tag{8}$$

where the Fokker–Planck operator $\hat{L}$ acting on $f(\mathbf{x}, t)$ has the form

$$\hat{L} f(\mathbf{x}, t) = -\frac{\partial}{\partial x_\ell} D_\ell^{(1)}(\mathbf{x}, t) f(\mathbf{x}, t) + \frac{\partial^2}{\partial x_\ell \partial x_{\ell'}} D_{\ell,\ell'}^{(2)}(\mathbf{x}, t) f(\mathbf{x}, t). \tag{9}$$

The functions $D_\ell^{(1)}(\mathbf{x}, t)$ and $D_{\ell,\ell'}^{(2)}(\mathbf{x}, t)$ are called Kramers–Moyal (KM) coefficients or drift and diffusion coefficients. Under the assumption of perfect knowledge of the trajectories $\mathbf{X}(t)$, the KM coefficients can be calculated from the incremental changes $\Delta X_\ell(t) \equiv X_\ell(t + \tau) - X_\ell(t)$ in an infinitesimal time interval τ as

$$D_\ell^{(1)}(\mathbf{x}, t) = \lim_{\tau \to 0} \frac{1}{\tau} \langle [\Delta X_\ell(t)] \rangle_{\mathbf{X}(t) = \mathbf{x}}, \tag{10a}$$

$$D_{\ell,\ell'}^{(2)}(\mathbf{x}, t) = \lim_{\tau \to 0} \frac{1}{2\tau} \langle [\Delta X_\ell(t)][\Delta X_{\ell'}(t)] \rangle_{\mathbf{X}(t) = \mathbf{x}}, \tag{10b}$$

where $\langle \ldots \rangle_{\mathbf{X}(t) = \mathbf{x}}$ denotes averages over the stochastic trajectories. The KM coefficients are related to the functions g_ℓ and $h_{\ell,\ell'}$ in the Langevin equation as

$$g_\ell(\mathbf{x}, t) = D_\ell^{(1)}(\mathbf{x}, t), \tag{11a}$$

$$h_{\ell,\ell'}(\mathbf{x}, t) = \sqrt{2 D_{\ell,\ell'}^{(2)}(\mathbf{x}, t)}. \tag{11b}$$

We consider only diagonal diffusion matrices, but the KM coefficients can depend explicitly on space and time. To estimate the coefficients, M-dimensional trajectories $X_{\ell,i}, \ell \in \{1, \ldots, M\}$ are sampled with a small, regular time step s at time points $i \in \{1, \ldots, N\}$. Trajectory samples $X_{\ell,i}$ are distinguished from the original stochastic variable $X_\ell(t)$ by the index i, representing the i-th time point. Therewith, two new sequences are constructed as

4

$$\mathbf{F}_\ell^{(1)} = \left\{ F_{\ell,i}^{(1)} \right\}_{i=1,\ldots,N} = \left\{ \frac{X_{\ell,i+1} - X_{\ell,i}}{s} \right\}_{i=1,\ldots,N}, \tag{12a}$$

$$\mathbf{F}_\ell^{(2)} = \left\{ F_{\ell,i}^{(2)} \right\}_{i=1,\ldots,N} = \left\{ \frac{(X_{\ell,i+1} - X_{\ell,i})^2}{2s} \right\}_{i=1,\ldots,N}, \tag{12b}$$

where s is a small time step[5]. The $\boldsymbol{F}_\ell^{(1)}$ and $\boldsymbol{F}_\ell^{(2)}$ are constructed with sample trajectories from random processes that are not differentiable. Use of these quantities for estimation of the KM coefficients in the spirit of Eq. (10) makes it necessary to first sample the stochastic process extensively to then approximate the average $\langle \ldots \rangle_{\mathbf{X}(t)=\mathbf{x}}$ over different realizations of the process.

Note that in basing the estimation on Eq. (10), we are neglecting two problems that occur for time series measured in the "real world". Firstly, measurement noise may render the assumption of a Markov process invalid on small scales[27]. Secondly, the finite sampling interval s cannot be made arbitrarily small in practice and therefore the estimated KM coefficients deviate systematically from the true coefficients[28,29]. Procedures for correcting finite-sampling-time errors are available for various stochastic processes[30–32]. While the focus of this work is on the inference problem for governing equations, finite sampling-time corrections should be employed in practical applications.

In the following, we employ two different methods for estimating the drift and diffusion coefficients. Firstly, a method is described in the next subsection that is based on binning of the data in phase space to produce histograms. Secondly, we compare the results obtained from data binning with results from direct estimation of the KM coefficients.

Estimation of KM coefficients from binned data. A classical method for the characterization of stationary, Markovian time series resulting from Langevin dynamics is based on binning of the trajectory data in space intervals[5,33,34]. For this approach, we focus on problems with only one space dimension ($M = 1$). To estimate probability distributions, the data from multiple sample trajectories of the stochastic process is grouped into Q bins and the values in each bin are averaged as

$$\{X_i\}_{i=1,\ldots,N} \mapsto \left\{ \bar{X}_k \right\}_{k=1,\ldots,Q} = \bar{\mathbf{X}}, \tag{13a}$$

$$\left\{ F_i^{(1)} \right\}_{i=1,\ldots,N} \mapsto \left\{ \bar{F}_k^{(1)} \right\}_{k=1,\ldots,Q} = \bar{\mathbf{F}}^{(1)}, \tag{13b}$$

$$\left\{ F_i^{(2)} \right\}_{i=1,\ldots,N} \mapsto \left\{ \bar{F}_k^{(2)} \right\}_{k=1,\ldots,Q} = \bar{\mathbf{F}}^{(2)}, \tag{13c}$$

where $\bar{X}_k$, $\bar{F}_k^{(2)}$, and $\bar{F}_k^{(2)}$ are bin-wise averages. The estimated probability for finding trajectory parts in the k-th bin, p_k, is normalized as $\sum_{k=1}^{Q} p_k = 1$ with $0 \leq p_k \leq 1$. Histograms resulting from data binning directly yield the curves for the drift and diffusion coefficients, see Refs.[5,33]. The equations for the KM coefficients, $D^{(1)}(x)$ and $D^{(2)}(x)$, are inferred by finding analytical expressions for $\bar{\mathbf{F}}^{(1,2)}$ as functions of $\bar{\mathbf{X}}$. For this purpose, a library $\boldsymbol{\Theta} \in \mathbb{R}^{Q \times K}$ is constructed from the binned data, where Q is the number of bins and K is the number of terms in the library. For example, $\boldsymbol{\Theta}(\bar{\mathbf{X}}) = [\mathbf{1}, \ \bar{\mathbf{X}}, \ \bar{\mathbf{X}} \odot \bar{\mathbf{X}}, \ \bar{\mathbf{X}} \odot \bar{\mathbf{X}} \odot \bar{\mathbf{X}}, \ \sin(\bar{\mathbf{X}}), \ \ldots]$ where $\odot$ again denotes an element-wise product. If the library contains all the function expressions that are necessary to describe the KM coefficients analytically, the governing equations can be written as

$$\bar{\mathbf{F}}^{(1)} = \boldsymbol{\Theta}(\bar{\mathbf{X}})\mathbf{W}^{(1)}, \tag{14a}$$

$$\bar{\mathbf{F}}^{(2)} = \boldsymbol{\Theta}(\bar{\mathbf{X}})\mathbf{W}^{(2)}, \tag{14b}$$

where $\mathbf{W}^{(1)}$ and $\mathbf{W}^{(2)}$ are two sparse vectors whose non-zero entries correspond to the library terms to be included in the sought-for analytical expressions for the KM coefficients. Equation (14a) yields $D^{(1)}(x)$ and Eq. (14b) yields $D^{(2)}(x)$. The inverse problems of finding optimal $\mathbf{W}^{(1,2)}$ in Eq. (14) have the same form as the problem in Eq. (2).

The binning of trajectories can produce significant errors in sparsely sampled regions, both in the interior and at the boundaries of the sampled phase space. We propose that the identification of SDEs can be improved by removal or filtering of the bins with high uncertainty. To substantiate this suggestion, we implement the inference procedure for unfiltered histograms and, additionally, implement a straight-forward extension that essentially consists of fixing a small probability threshold, below which all the data is discarded. While the probability threshold can can be determined in different ways, we employ here an automatic heuristic that was originally designed for edge detection in images[35]. The procedure that is described in Ref.[35] consists of dividing the data according to probability thresholds to maximize the Shannon and Tsallis entropy, respectively. Maximization of the Shannon entropy produces thresholds that divide the data into "foreground" and "background", corresponding to

signal-dominated and noise-dominated phase-space regions, respectively. The threshold value determining the "background" is then improved in a second step by maximizing the Tsallis entropy, whose pseudo additivity reportedly improves the analysis of data containing long-range correlations, see also Ref.[36]. While we found that this method for determining a probability threshold is useful in practice, its theoretical underpinnings are to our knowledge not entirely clear. Thus, a manual selection of the probability threshold based on the results may be preferable in some cases.

Estimation of KM coefficients without data binning. A more direct approach for estimating the KM coefficients is based on the use of the trajectories $\mathbf{F}_\ell^{(1)}$ and $\mathbf{F}_\ell^{(2)}$ without binning or filtering. Since we do not intend to study transient initial dynamics, we mostly employ as input data a single, long trajectory generated from the stochastic process. For inference of the KM coefficients from the space-time trajectories, we construct a library $\boldsymbol{\Theta} \in \mathbb{R}^{N \times K}$, where N is the length of the trajectory and K is the number of terms in the library. For example, $\boldsymbol{\Theta}(\{\mathbf{X}_{\ell'}\}) = [\mathbf{1},\ \mathbf{X}_1,\ \dots,\mathbf{X}_M,\ \mathbf{X}_1 \odot \mathbf{X}_2,\ \dots,\ \sin(\mathbf{X}_1),\dots]$, where ℓ' covers all M components of the stochastic process. Note that the library is constructed such that $F_{\ell,i}^1$ and $F_{\ell,i}^2$ at the i-th time point depend only on functions involving coordinates $\{X_{\ell',i}\}_{\ell'}$ at the same time point. Thus, a velocity dependence or a history dependence of the estimators for the drift and diffusion coefficients is excluded. Under the assumption that the library contains all necessary terms describing the drift and diffusion coefficients, the coefficients for the ℓ-th component of the stochastic process can be inferred with

$$\mathbf{F}_\ell^{(1)} = \boldsymbol{\Theta}(\{\mathbf{X}_{\ell'}\})\mathbf{W}_\ell^{(1)}, \tag{15a}$$

$$\mathbf{F}_\ell^{(2)} = \boldsymbol{\Theta}(\{\mathbf{X}_{\ell'}\})\mathbf{W}_\ell^{(2)}, \tag{15b}$$

where $\ell \in \{1 \dots M\}$ and the vectors $\mathbf{W}_\ell^{(1,2)}$ are non-zero in those entries that correspond to the terms in the libary that are required for the analytical description of the KM coefficient. The determination of the $\mathbf{W}_\ell^{(1,2)}$ is again an inverse optimization problem.

Solution of the inference problems with automatic threshold sparse Bayesian learning.

For identification of the relevant library terms as, e.g., for Eq. (15), we propose a method that we call automatic threshold sparse Bayesian learning (ATSBL). The method consists of two main steps. First, the inverse problem is solved with an efficient algorithm called Bayesian compressive sensing using Laplace priors (BCSL)[37]. Since the library is large, the solution vector generated by the BCSL algorithm typically still contains quite a few non-vanishing but small entries. Therefore, in a second step, the negligible contributions to the resulting governing equations are removed by an automatic thresholding procedure[3,5,7]. These two steps of the method are detailed below.

Bayesian compressive sensing using Laplace priors (BCSL). We consider a generic linear equation system involving a given vector $\mathbf{g}$ and matrix $\boldsymbol{\Phi}$ and an unknown, sparse vector $\mathbf{w}$ as

$$\mathbf{g} = \boldsymbol{\Phi}\mathbf{w} + \mathbf{s}, \tag{16}$$

where the vector $\mathbf{s}$ represents noise or measurement errors. Here, $\mathbf{w}$ can be thought of as a solution vector appearing in an iterative solution procedure for Eqs. (5), (14), or Eq. (15). Various methods can be used to calculate sparse solution vectors $\mathbf{w}$ from Eq. (16). In particular research on compressive sensing, which deals with the reconstruction of sparse signals from underdetermined systems, has yielded broadly applicable, efficient methods for finding sparse solution vectors $\mathbf{w}$. Among these are Bayesian methods based on the relevance vector machine (RVM)[38,39]. Very sparse result vectors are obtained if a Laplace distribution is used as a prior probability distribution for $\mathbf{w}$. Here, we employ a method called Bayesian compressive sensing using Laplace priors (BCSL)[37]. Specifically, we employ a variant of BCSL that interatively calculates approximate solutions, which is very computationally efficient and yields accurate results for our type of applications.

Briefly, the mathematical basis of BCSL is as follows, see Ref.[37]. The method is based on a three-stage hierarchical model. It is assumed that the errors $\mathbf{s}$ are drawn from a zero-mean Gaussian distribution with unknown variance $1/\beta > 0$. Therefore, the likelihood function for finding a vector $\mathbf{g}$ is given by

$$p(\mathbf{g}|\mathbf{w}, \beta) = \left(\frac{2\pi}{\beta}\right)^{-\frac{N}{2}} \exp\left\{-\frac{\beta}{2}\|\mathbf{g} - \boldsymbol{\Phi}\mathbf{w}\|_2^2\right\}. \tag{17}$$

The unknown vector $\mathbf{w}$ is assigned a prior distribution, which represents our knowledge on the nature of this quantity. To encode sparsity, one would like to employ a Laplace prior $p(\mathbf{w}|\lambda) = \lambda/2 \exp(-\lambda \sum_i |w_i|/2)$ with a

6

hyperparameter λ. However, the evaluation of integrals using this choice of a Laplace prior is not readily achieved since the Laplace prior is not conjugate to the Gaussian likelihood, Eq. (17). Therefore, an auxiliary vector of non-negative hyperparameters $\boldsymbol{\gamma}$ with the same dimension as $\mathbf{w}$ is employed to express the prior as the convolution of the two different distributions $p(\mathbf{w}|\boldsymbol{\gamma}) = \Pi_i \left[\exp\left(-w_i^2/(2\gamma_i)\right)/\sqrt{2\pi\gamma_i} \right]$ and $p(\boldsymbol{\gamma}|\lambda) = \Pi_i \left[\lambda \exp\left(-\lambda\gamma_i/2\right)/2 \right]$. These two distributions together result in a Laplace prior after marginalizing out $\boldsymbol{\gamma}$ as

$$p(\mathbf{w}|\lambda) = \int p(\mathbf{w}|\boldsymbol{\gamma})p(\boldsymbol{\gamma}|\lambda)\mathrm{d}\boldsymbol{\gamma} = \frac{\lambda^{N/2}}{2^N}e^{-\sqrt{\lambda}\sum_i |w_i|}, \tag{18}$$

see Ref.[40]. Overall, the joint probability density results as

$$p(\mathbf{g}, \mathbf{w}, \boldsymbol{\gamma}, \lambda, \beta) = p(\mathbf{g}|\mathbf{w}, \beta)p(\mathbf{w}|\boldsymbol{\gamma})p(\boldsymbol{\gamma}|\lambda)p(\lambda)p(\beta), \tag{19}$$

where the parameters λ and β are both assumed to obey Gamma distributions. To infer values for the most probable solution vector $\mathbf{w}$ as well as the hyperparameters, an evidence procedure is employed wherein the posterior probability $p(\mathbf{w}, \boldsymbol{\gamma}, \lambda, \beta|\mathbf{g})$ is maximized with respect to $\mathbf{w}, \boldsymbol{\gamma}, \lambda$, and β, given the data. By making use of the expression

$$p(\mathbf{w}, \boldsymbol{\gamma}, \lambda, \beta|\mathbf{g}) = p(\mathbf{w}|\mathbf{g}, \boldsymbol{\gamma}, \lambda, \beta)p(\boldsymbol{\gamma}, \lambda, \beta|\mathbf{g}) = \frac{p(\mathbf{g}, \mathbf{w}, \boldsymbol{\gamma}, \lambda, \beta)}{p(\mathbf{g})} \tag{20}$$

together with Eq. (19), one sees that the value of $\mathbf{w}$ that maximizes the posterior can be determined by simply maximizing $p(\mathbf{g}|\mathbf{w}, \beta)p(\mathbf{w}|\boldsymbol{\gamma})$. This calculation yields for the result vector the expression $\mathbf{w}^* = \beta \boldsymbol{\Sigma} \boldsymbol{\Phi}^T \mathbf{g}$ with $\boldsymbol{\Sigma} = (\beta \boldsymbol{\Phi}^T \boldsymbol{\Phi} + \boldsymbol{\Lambda})^{-1}$ and $\boldsymbol{\Lambda} = \mathrm{diag}(1/\gamma_i)$. This step corresponds to a Ridge regression that depends on the unknown values of $\boldsymbol{\gamma}, \lambda$, and β. Determination of these hyperparameters proceeds by maximizing

$$p(\boldsymbol{\gamma}, \lambda, \beta|\mathbf{g}) = \frac{p(\boldsymbol{\gamma}, \lambda, \beta, \mathbf{g})}{p(\mathbf{g})} \tag{21}$$

with respect to $\boldsymbol{\gamma}, \lambda$, and β. Here, $p(\boldsymbol{\gamma}, \lambda, \beta, \mathbf{g})$ is calculated from the right hand side of Eq. (19) by integrating out $\mathbf{w}$. With the fast, approximate version of BCSL, the equations determining the optimal values of the hyperparameters are solved iteratively, where only one entry of the vector $\boldsymbol{\gamma}$ is adjusted in every step.

Automatic thresholding. Solution of the inverse problem (16) with BCSL typically yields vectors $\mathbf{w}$ that contain only a few large entries, but also a number of very small, non-zero entries. Removal of these negligible entries is desirable and we improve the solution sparsity with an iterative thresholding procedure[4]. The pseudocode 1 illustrates how the thresholding procedure proposed in in Ref.[4] is combined with BCSL proposed in Ref.[37]. Briefly, the thresholding algorithm works as follows. The input is given by $\mathbf{g}$, the library matrix $\boldsymbol{\Theta}$, an initial increment d_{tol} for the threshold *tol*, and the maximum number of iterations n_{iters}. The data $\mathbf{g}$ and $\boldsymbol{\Theta}$ is spilt into two parts for training and test, respectively. Usually, 80% of the data is used for training and 20% for testing. Thresholds are calculated iteratively from the training data and the validity of the thresholds is evaluated based on the error resulting from their application to the test data. The core part of the algorithm is a loop for iterative calculation of the sparse vector $\mathbf{w}$ and the threshold *tol*. In each iteration step, the approximate, fast BCSL routine is first employed to obtain an estimate of $\mathbf{w}$ from the training data. The quality of this solution estimate is evaluated by calculating the resulting error with the test data

$$e = \|\boldsymbol{\Theta}^{\mathrm{test}}\mathbf{w} - \mathbf{g}^{\mathrm{test}}\|_2^2 + \eta\|\mathbf{w}\|_0, \tag{22}$$

where the penalty factor of the solution norm is chosen to depend on the condition number as $\eta = 10^{-3}\kappa(\boldsymbol{\Theta})$, as suggested for the original algorithm[4]. If the error of the current solution is smaller than the error of previous iterations, the new solution is accepted and the threshold *tol* is increased. In the opposite case, the threshold is decreased and the increment d_{tol} is refined. The final solution $\mathbf{w}_{\mathrm{best}}$ is the sparse vector that determines the terms in the governing differential equations, SDEs, ODEs, and PDEs.

Function: $\mathrm{ATSBL}(\boldsymbol{\Theta}, \mathbf{g}, d_{\mathrm{tol}}, n_{\mathrm{iters}})$
%% Split data into parts for training and test
$\boldsymbol{\Theta} \mapsto [\boldsymbol{\Theta}^{\mathrm{train}}, \boldsymbol{\Theta}^{\mathrm{test}}]$ % ca. 80% training, 20% test
$\mathbf{g} \mapsto [\mathbf{g}^{\mathrm{train}}, \mathbf{g}^{\mathrm{test}}]$

%% Initialization
$\sigma^2 = \mathrm{var}(\mathbf{g}^{\mathrm{train}})$ % Variance of data.
$\eta = 10^{-3}\kappa(\boldsymbol{\Theta})$ % $\kappa(\boldsymbol{\Theta})$ is condition number
$tol = d_{\mathrm{tol}}$ % Initial threshold
$Q = \mathrm{size}(\boldsymbol{\Theta}, 2)$ % Number of library terms
$\mathbf{w}_{\mathrm{best}} = (\boldsymbol{\Theta}^{\mathrm{train}})^{-1}\mathbf{g}^{\mathrm{train}}$ %Initial solution guess
$e_{\mathrm{best}} = \|\boldsymbol{\Theta}^{\mathrm{test}}\mathbf{w}_{\mathrm{best}} - \mathbf{g}^{\mathrm{test}}\|_2^2 + \eta\|\mathbf{w}_{\mathrm{best}}\|_0$ %Initial error

%% Solution with iterative threshold adaptation
for $(\ i = 1;\ i < n_{\mathrm{iters}};\ i = i + 1\ )$ {
    $\hat{\mathbf{w}} = \mathrm{FastBCSL}(\boldsymbol{\Theta}^{\mathrm{train}}, \mathbf{g}^{\mathrm{train}}, \sigma^2)$
    bigcoeffs $= \{m : |\hat{w}_m| \geq tol\}$
    $\boldsymbol{\Theta}_{\mathrm{old}}^{\mathrm{train}} = \boldsymbol{\Theta}^{\mathrm{train}}$;
    $\boldsymbol{\Theta}^{\mathrm{train}} = \boldsymbol{\Theta}^{\mathrm{train}}(:, \mathrm{bigcoeffs})$
    if $i == 1$ **then**
        $final_p = \mathrm{bigcoeffs}$;
        $\mathbf{w} = \hat{\mathbf{w}}$
    else
        $final_p = final_p(\mathrm{bigcoeffs})$
        $\mathbf{w} = \mathrm{zeros}(Q, 1)$
        $\mathbf{w}(final_p) = \hat{\mathbf{w}}(\mathrm{bigcoeffs})$
    end

    % Calculate error and threshold
    $e = \|\boldsymbol{\Theta}^{\mathrm{test}}\mathbf{w} - \mathbf{g}^{\mathrm{test}}\|_2^2 + \eta\|\mathbf{w}\|_0$
    % Adapt threshold
    if $e \leq e_{best}$ **then**
        % Error is decreasing. Increase threshold
        $e_{\mathrm{best}} = e$
        $\mathbf{w}_{\mathrm{best}} = \mathbf{w}$
        $tol = tol + d_{\mathrm{tol}}$
    else
        % Tolerance too high
        $\boldsymbol{\Theta}^{\mathrm{train}} = \boldsymbol{\Theta}_{\mathrm{old}}^{\mathrm{train}}$
        $tol = \mathrm{max}([0, tol - 2d_{\mathrm{tol}}])$
        % Change threshold stepping
        $d_{\mathrm{tol}} = \frac{2d_{\mathrm{tol}}}{n_{\mathrm{iters}}-i}$
        $tol = tol + d_{\mathrm{tol}}$
    end
}
return $\mathbf{w}_{\mathrm{best}}$

Algorithm 1: Pseudocode for automatic threshold sparse Bayesian learning (ATSBL), which combines BCSL [37] and TrainSTRidge [4] to achieve parameter-free inference of highly sparse solutions to inverse problems.

Quality score for identified governing equations.   The error of the inference procedure can be directly quantified by comparison of the results with a known set of original differential equations in test cases. For this purpose, we define the deviation of identified coefficient (DIC) as

$$\mathrm{DIC} = \frac{1}{K} \sum_{\{i|(w_i \neq 0 \vee w'_i \neq 0)\}} \frac{|w_i - w'_i|}{\max(|w_i|,\ |w'_i|)}, \tag{23}$$

where every w_i is a coefficient of one term in the identified equation and w'_i is the related coefficient in the original equation that was used to generate the test data. Here, at least one of the coefficients in each pair $\{w_i, w'_i\}$ is required to be non-zero and the sum only runs over these coefficients. K represents the number of these coefficients. The DIC lies in the range $[0, \infty]$ where 0 indicates a perfectly identified equation.
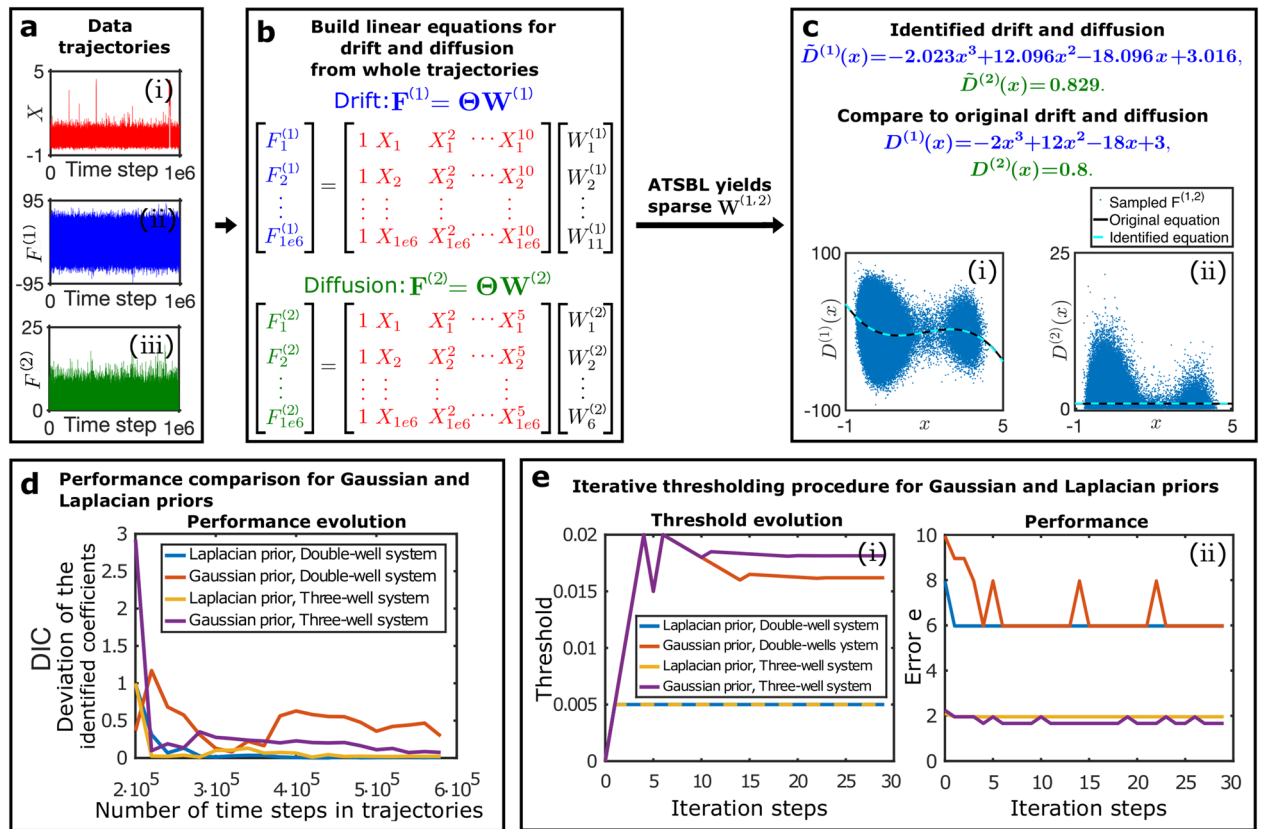
Figure 1. Data-driven discovery of a one-dimensional SDE with automatic threshold sparse Bayesian learning (ATSBL). (**a-i**) Trajectory of a particle undergoing overdamped diffusive motion in a double-well potential (10^6 time steps). (**a-ii,a-II**) Values of the $\mathbf{F}^{(1)}$ and $\mathbf{F}^{(2)}$ generated with discrete differences from the same trajectory. (**b**) The library matrix Θ is constructed by evaluating a given set of functions for all values of the trajectory. Thereby, one obtains linear equation systems that relate the known sequences $\mathbf{F}^{(1,2)}$ to unknown, sparsely populated coefficient vectors $\mathbf{W}^{(1,2)}$. The determination of the non-zero entries of $\mathbf{W}^{(1,2)}$ yields a set of library functions that together describe the drift and diffusion coefficients $D^{(1)}$ and $D^{(2)}$. (**c**) Exemplary results of the inference procedure. Despite the large noise amplitude, accurate predictions can be made directly from the trajectory data. (**d**) Comparison of the use of a Laplacian and Gaussian prior distribution in the inference procedure. The deviation of the identified coefficient (DIC) for the drift coefficient is plotted against the number of data points used for training. The Laplace prior in ATSBL decreases the error and reduces the required sample size. (**e**) Convergence rate of the thresholding procedure for Laplacian and Gaussian prior distributions. (**e-i**) Laplace priors result in fast threshold convergence. (**e-ii**) The error e defined in Eq. (22) decreases during the iterations. Errors achieved with Gaussian- and Laplacian priors are comparable.

Results

Inference of SDEs from noisy trajectories.
We first illustrate data-driven identification of SDEs by the example of overdamped Brownian motion of a particle inside a one-dimensional double-well potential with coordinate x. The drift and diffusion coefficients of this system are given by

$$D^{(1)}(x) = -2x^3 + 12x^2 - 18x + 3, \tag{24a}$$

$$D^{(2)}(x) = 0.8. \tag{24b}$$

The trajectory data that is to be used for inferring the governing equation is generated by integrating the Langevin equation with the Euler-Maruyama method. A trajectory $\mathbf{X}$ is shown in Fig. 1a-i (10^6 time steps). The trajectories $\mathbf{F}^{(1)}$ and $\mathbf{F}^{(2)}$ are shown in Fig. 1a-ii,iii. To visualize the x- dependence of the estimator for the drift coefficient, we plot $\mathbf{F}^{(1)}$ against $\mathbf{X}$, see Fig. 1c-i. Similarly, $\mathbf{F}^{(2)}$ is plotted against $\mathbf{X}$ as estimator of the diffusion coefficient $D^{(2)}(x)$ in Fig. 1c-ii. Both plots exhibit large fluctuations around the true drift and diffusion coefficients and the resulting averages are clearly prone to errors, particularly at the boundaries of the sampled domain.

Using the trajectory data, we next construct a library consisting of 11 terms for the drift coefficient and 6 terms for the diffusion coefficient as illustrated in Fig. 1b. Then, we employ ATSBL to identify $\mathbf{W}^{(1)}$ and $\mathbf{W}^{(1)}$ directly from the trajectory without binning. The identified x-dependent functions for the drift and diffusion coefficients are plotted in Fig. 1c-i,ii. They agree well with the original functions used for creating the data. The identified equations with estimated uncertainties are shown in Fig. 1c.
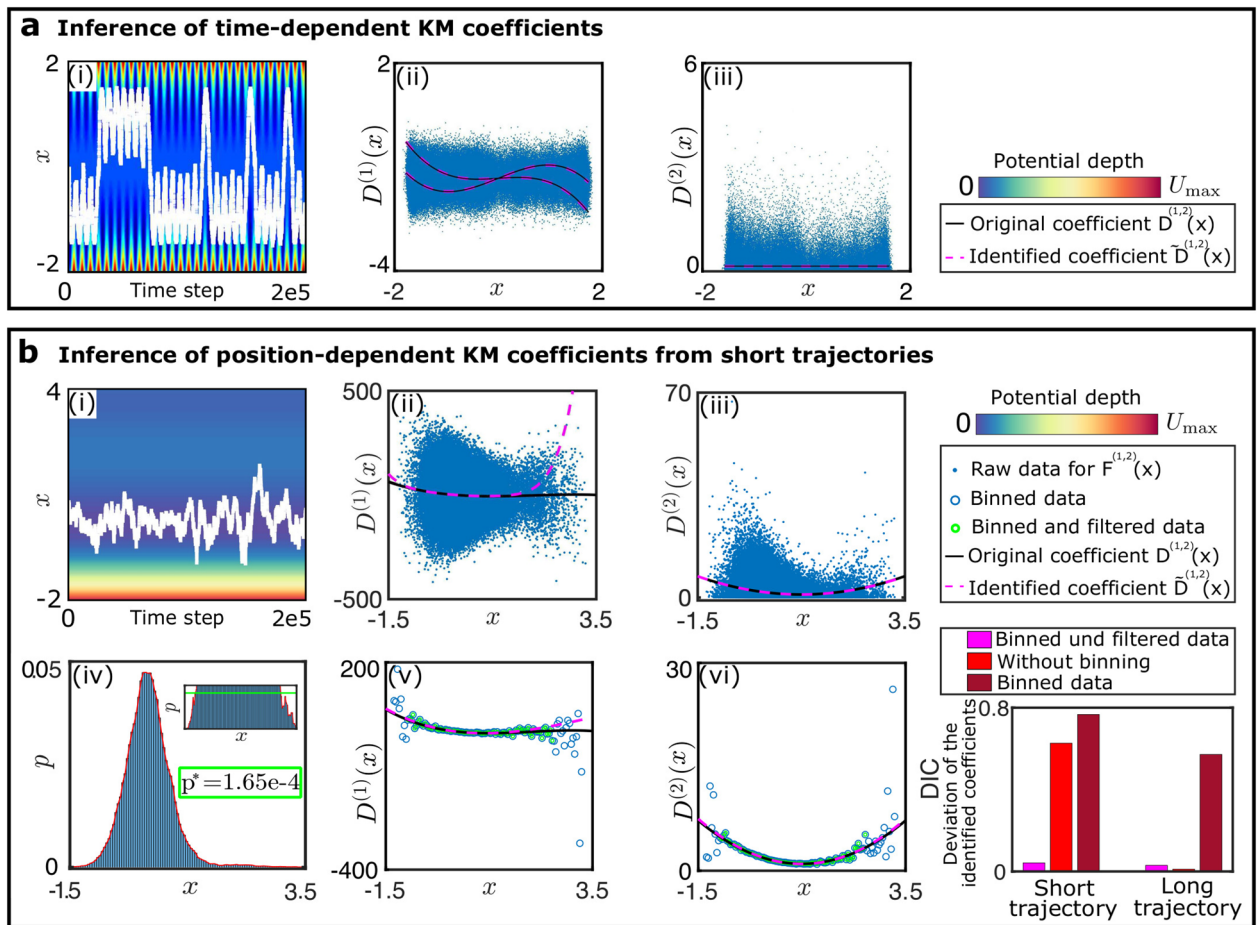
Figure 2. Inference of KM coefficients for one-dimensional SDEs. (**a**) System with a time-dependent force field. (**a-i**) Trajectory of an overdamped motion in a time-varying double-well potential. (**a-ii,iii**) Using an appropriate function library, the functional forms of the KM coefficients can be faithfully reconstructed. Blue dots are values of $F^{(1,2)}$ estimated from the trajectory. (**b**) Advantage of data binning for analysis of short trajectories. (**b-i**) Trajectory resulting from overdamped motion in a double-well potential with space-dependent diffusion coefficient. (**b-ii,iii**) Inferred x-dependence of the KM coefficients for short trajectories (2×10^5 time steps). The unpopulated regions in phase space are characterized by a high uncertainty of inference and therefore lead to large deviations in the coefficients. (**b-iv**) Histogram of particle positions for the trajectory shown in (**i**). (**b-iv,v,vi**) Binned distributions can be used to infer the KM coefficients, but large errors occur in regions that are not well-sampled. Inference errors due to incomplete phase-space sampling for short trajectories can be accounted for by excluding the data below a probability threshold, corresponding to large uncertainty. (**b-vii**) Performance of the inference with data binning and without data binning for short and long trajectories (2×10^5 and 2×10^7 time steps, respectively). The shown DIC is the average of the DICs for $D^{(1)}$ and $D^{(2)}$. For long trajectories, data binning does not reduce the error.

The main distinction of ATSBL as compared to established inference techniques is the assumption of a Laplacian distribution for the prior of the library coefficients. The more direct, albeit theoretically less sparsity-promoting procedure is to employ a Gaussian prior, corresponding to a ridge regression with fixed regularization parameter, for inference of the solution vector **w** in Eq. (16) prior to automatic thresholding, as done, e.g., in Ref.[4]. To compare the performance of these two approaches for inference of SDEs, we evaluate the deviation of the identified coefficients, DIC, as a function of the number of data points used for inference. The results shown in Fig. 1d indicate that the Laplacian prior is preferable over the Gaussian prior since it requires less data and results in a smaller DIC. To further establish the robustness of ATSBL, we consider the convergence of the iterative thresholding procedure for each of the two prior distributions. The result shown in Fig. 1e-i,ii demonstrate a better convergence achieved in the case of the Laplacian prior. For both, Gaussian and Laplacian prior distributions, the threshold and the error oscillate during the iteration process, which is due to the adaptive step size during the thresholding.

Inference of SDEs with time-dependent drift coefficient.

In the previous section, an example is provided of how the KM coefficients can be obtained by performing a regression directly with the trajectory data. The direct use of the trajectory data becomes particularly important for the treatment of the more complex situation of a time-varying force. In such a situation, the probability distributions change over time and a histogram-

based approximation of the dynamic distributions can be technically challenging and requires the availability of many sample trajectories for the same conditions. In order to explore the validity of our approach in this situation, we consider the example of a particle diffusing within a time-dependent one-dimensional potential. The drift and diffusion coefficients of this system are given by

$$D^{(1)}(x) = [a_0 + 1 - \cos(\omega t)]x - x^3, \tag{25a}$$

$$D^{(2)}(x) = 0.8, \tag{25b}$$

with $a_0 = 5 \times 10^{-3}$. The potential has a double-well shape, where the positions of the two minima vary in time. The two minima start at separate positions and merge periodically into one minimum before separating again to reach the initial positions. Each time the potential wells come close to each other, the transition probability becomes large and the particle is likely to change from one well to another. This gives rise to stochastic oscillations between the two potential wells. To test whether the underlying equations can be inferred with a library constructed from a single trajectory, we assume that the frequency ω at which the potential changes is known. Inference of this frequency from the data is in principle also possible, but requires excessive computational power since high-order terms with explicit time-dependence must be accounted for in the library. We construct a library consisting of a time-dependent and a time-independent part. The first half of the library is simply a polynomial basis, the second half corresponds to the polynomial basis multiplied with a $\cos(\omega t)$ factor. The results of the inference procedure are shown in Fig. 2a. The inferred equation is in agreement with the correct equation. For illustration, we plot snapshots of the drift and diffusion for the original equations and the inferred equations in Fig. 2a-ii,iii. Note that the inference procedure for first-order SDEs shows a remarkable performance even though only one sample trajectory is used for inference.

Data binning for inference of SDEs from short trajectories.   An inference method that relies on direct use of sample trajectories for a regression can become unreliable when confronted with short trajectories in an inhomogeneous force field. In such a situation, we find that it is more appropriate to employ data binning. We illustrate this procedure with Brownian motion of a particle in a one-dimensional double-well potential where the diffusion coefficient depends on space. The drift and diffusion coefficients of the model are given by

$$D^{(1)}(x) = -2x^3 + 12x^2 - 18x + 3, \tag{26a}$$

$$D^{(2)}(x) = x^2 - 2x + 2. \tag{26b}$$

We first consider short trajectories that have 2×10^5 time steps, exemplified by the plot in Fig. 2b-i. The raw data and the binned data are shown in Fig. 2b-ii,iii and b-v,vi, respectively. For this example, 200 data bins are employed. The distributions approximated by the binned data clearly deviate from the known functions $D^{(1)}(x)$ and $D^{(2)}(x)$ in undersampled regions. Therefore, the binned data is filtered to remove data points with high uncertainty. This filtering is done as described in the "Methods" section by discarding bins below a probability threshold p^* that is determined by entropy maximization[35], see Fig. 2b-iv. To assess if binning and filtering is also beneficial for inference of SDEs from long trajectories, we also use data from trajectories with 2×10^7 time steps. Figure 2b-vii shows the errors of the identified coefficients.

For short trajectories, binning is advantageous in combination with a filtering procedure to suppress data with high uncertainty. The reason for this result can be understood from inspection of Fig. 2b-v,vi, where the inferred functions match the correct functions only in the most populated regions of phase space. Thus, the exclusion of data points with high uncertainty prevents overfitting and improves the inference of the underlying dynamical equations if the trajectory is not long enough to allow a sufficient sampling of the whole phase space. Conversely, data binning with or without filtering is disadvantageous for the analysis of long trajectories that sample the whole phase space, see Fig. 2b-vii.

Active sampling improves the identification of SDEs.   We have so far restricted our attention to the extraction of estimates from data that was generated prior to the analysis, e.g., in experiments. Thereby, we have assumed that the size of the data set is large enough to allow some form of inference of the governing equations. In a different scenario one might have the ability to perturb the studied system, either in a computer simulation or in an experimental setup, while simultaneously recording the data. Then, one may enhance the sampling efficiency by means of an appropriately designed perturbation that is applied to the system. Generally, this methodology is expected to be useful whenever the system exhibits an energy landscape with multiple local minima that can trap the trajectory for long times. We describe an adaptive control method where the inference of the dynamical equations together with a simultaneous perturbation of the system recursively results in a global exploration of the phase space to provide sufficient sampling everywhere.

Since the probability distribution tends to be peaked around local energy minima, the dynamical equations can be estimated locally near these minima. To take advantage of this local estimation while iteratively extending the sampled region, we re-sample repeatedly while applying in each sampling round a control force that is opposite to a force from the system that is estimated locally from previous rounds. The difficulty with a straight-forward application of this method is that the control force can admit large deviations away from the initial estimation region. This effect produces large errors, slows down convergence, and may even lead to divergence problems. We overcome this problem by weighting the control force with a Gaussian distribution, such that the control force vanishes away from the current estimation region. This local control force expels the
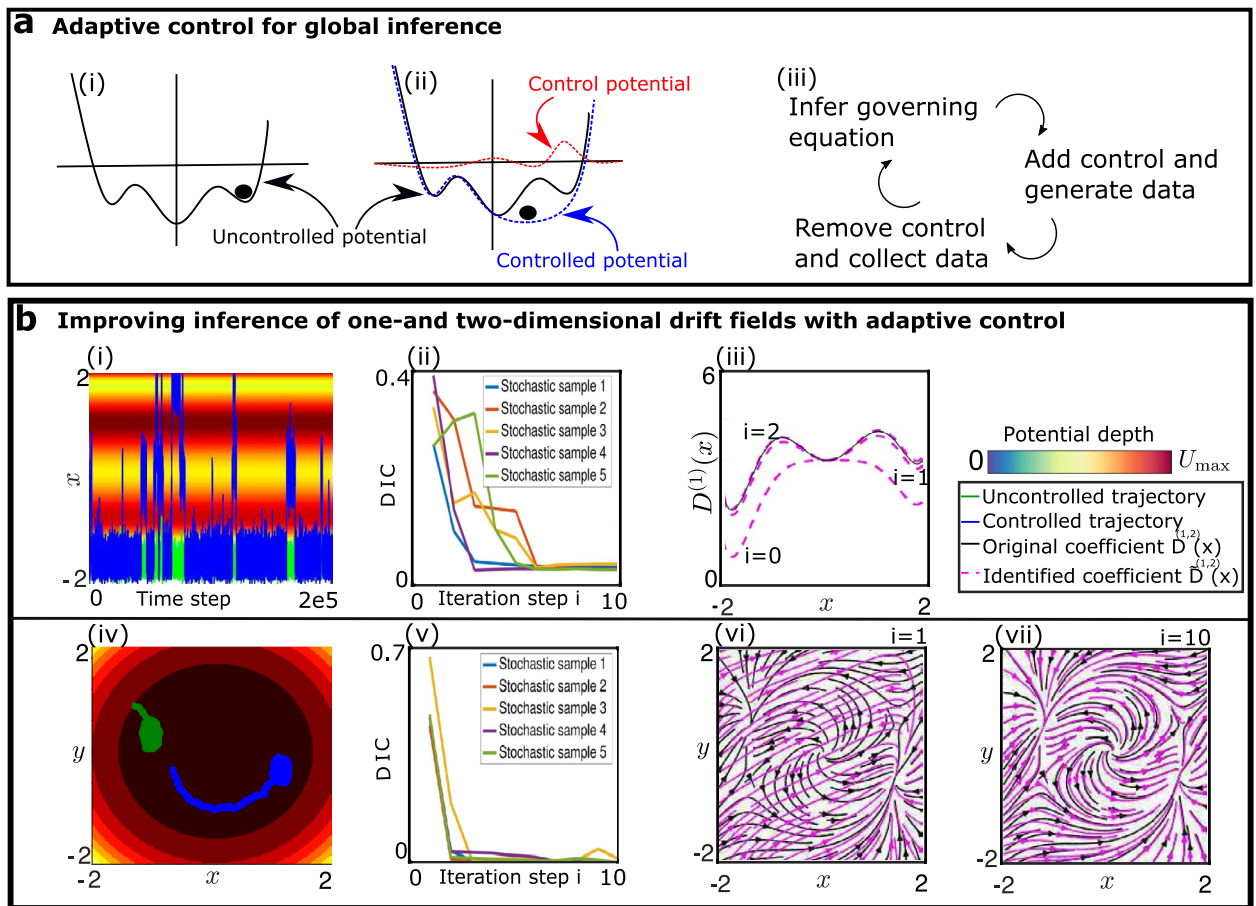
Figure 3. Active learning with automatic iterative sampling optimization (AISO). (**a**) Schematic presentation of automatic iterative sampling optimization for the case of Brownian diffusion (**a-i**). Initially, the particle is trapped in a local energetic minimum and the functional form of the potential can therefore only be inferred locally. (**a-ii**) After the first iteration step, the potential hypersurface near the estimated minimum is flattened and the particle can thus explore other regions of phase space. The same procedure is repeated iteratively and the control is always applied at the minimum estimated during the previous iteration. (**a-iii**) Schematic representation of the main feedback control loop. (**b-i**) Trajectory of a particle undergoing Brownian motion in a one-dimensional three-well potential. The green curve shows a trapped trajectory while the blue curve shows a trajectory in the presence of control forces. (**b-ii**) The deviation of the inferred coefficients (DIC) decreases during the iterations. (**b-iii**) The identified drift field converges to the correct function during the iteration. (**b-iv**) Trajectory of a particle undergoing diffusion in a two-dimensional force field. The green curve exemplifies a trapped trajectory for plain sampling. The blue curve shows an example of a trajectory in the presence of control forces. The color of the background only represents part of the force field, namely a Mexican hat potential $V(x, y) = -(x^2 + y^2)/2 + (x^2 + y^2)^2/4$ that generates radial forces. (**b-v**) Evolution of the of the DIC during the iterations. (**b-vi**) Streamlines of the identified drift field (pink) and streamlines of the correct drift field (black) after the first iteration step. (**b-vii**) Streamlines of identified drift field (pink) and streamlines of the correct drift field (black) after the tenth iteration step. The identified force field at the end of the iteration closely matches the original one.

trajectory from the energy minimum where the estimation has been performed and the trajectory eventually reaches another local minimum.

The method, which we call automatic iterative sampling optimization (AISO), is illustrated in Fig. 3a,i–iii and the pseudocode is provided in Algorithm 2. At each iteration, the underlying dynamical equations are estimated from the data accumulated during all previous iterations. The negative of the inferred drift term is employed locally as control force. The center and width of the Gaussian weight of the control force is calculated only from the mean and standard deviation of the trajectory of the previous step. Thus, we define our control force acting on the component ℓ as

$$c_\ell^i(\{X_{\ell'}(t)\}) = -\boldsymbol{\Theta}_\ell(\{X_{\ell'}(t)\}) \cdot \mathbf{w}^i \exp\left[-\frac{\| X_\ell(t) - \mu_\ell^i \|^2}{\zeta_\ell^i} \right] \tag{27}$$

where the index i indicates that values are to be taken at the iteration number i; μ_ℓ^i and ζ_ℓ^i stand for the mean and variance of the trajectory extracted from the step i in each iteration. After a sufficient number of iterations,
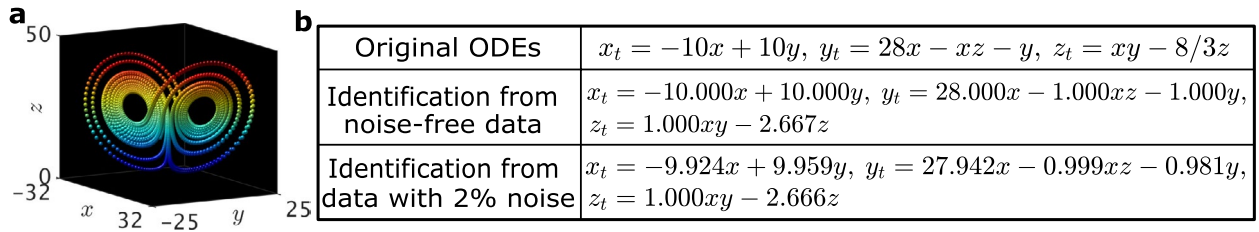
Original ODEs	$x_t = -10x + 10y, \; y_t = 28x - xz - y, \; z_t = xy - 8/3z$
Identification from noise-free data	$x_t = -10.000x + 10.000y, \; y_t = 28.000x - 1.000xz - 1.000y,$ $z_t = 1.000xy - 2.667z$
Identification from data with 2% noise	$x_t = -9.924x + 9.959y, \; y_t = 27.942x - 0.999xz - 0.981y,$ $z_t = 1.000xy - 2.666z$

Figure 4. Example for data-driven discovery of ODEs with ATSBL. (**a**) Plot of a numerically integrated trajectory for $t \in [0, 25]$ with a time step of $\Delta t = 2 \times 10^{-4}$ and an initial condition as $[x_0, \; y_0, \; z_0] = [-8, \; 8, \; 27]$. (**b**) The table shows the original ODEs, i.e., the Lorenz system, and the identified ODEs from noise-free data and data with 2% Gaussian noise.

the data points accumulated from all iterations are combined and the equation of motion is extracted from the accumulated data. This procedure is repeated for a predefined number of iteration steps. For the examples presented in the following, the iteration step number has been fixed to $N = 10$, since convergence has been achieved within less than 10 steps in these cases.

For a first demonstration of our method, we employ a three-well potential $U(x) = x^6 - 6x^4 + 0.5x^3 + 8x^2$ with a constant diffusion coefficient for simulating the trajectory of a particle in one dimension. The drift and diffusion coefficients are

$$D^{(1)}(x) = -\frac{dU}{dx} = -6x^5 + 24x^3 - 1.5x^2 - 16x, \tag{28a}$$

$$D^{(2)}(x) = 1. \tag{28b}$$

Next, we also consider a two-dimensional drift field, consisting of a radially symmetric component and a shear component in the x, y-plane. The drift and diffusion coefficients are given by

$$D_x^{(1)}(x, y) = x(1 - x^2 - y^2) + y(x^2 - y^2 - b) \tag{29a}$$

$$D_y^{(1)}(x, y) = y(1 - x^2 - y^2) + x(x^2 - y^2 - b), \tag{29b}$$

$$D_x^{(2)}(x, y) = D_y^{(2)}(x, y) = 1. \tag{29c}$$

Using these driving forces, we simulate trajectories with 10^5 time steps with one time step being $\Delta t = 5 \times 10^{-3}$. Parts of the trajectories on the potential maps are shown in Fig. 3b-i,iv. Results for the intermediate iteration steps are shown together with the drift field in Fig. 3b-iii,vi. As the algorithm proceeds through more iterations, the coefficients of the control potential approach the coefficients of the correct drift field, and the expulsion from each local minimum becomes more efficient, Fig. 3b-iii,vii. This results in an enhancement of rare events where the particle crosses the saddle points, as illustrated in Fig. 3b-i,iv by the controlled and uncontrolled trajectories. The error is quantified by calculating the coefficients $\tilde{D}^{(1)}(x)$ and $\tilde{D}^{(2)}(x)$ in each iteration. The DIC reduces from 1 to nearly 0.01 during the iterations, see Fig. 3b-ii,v. Thus, the terms of the identified equations approach those given in the original equations.
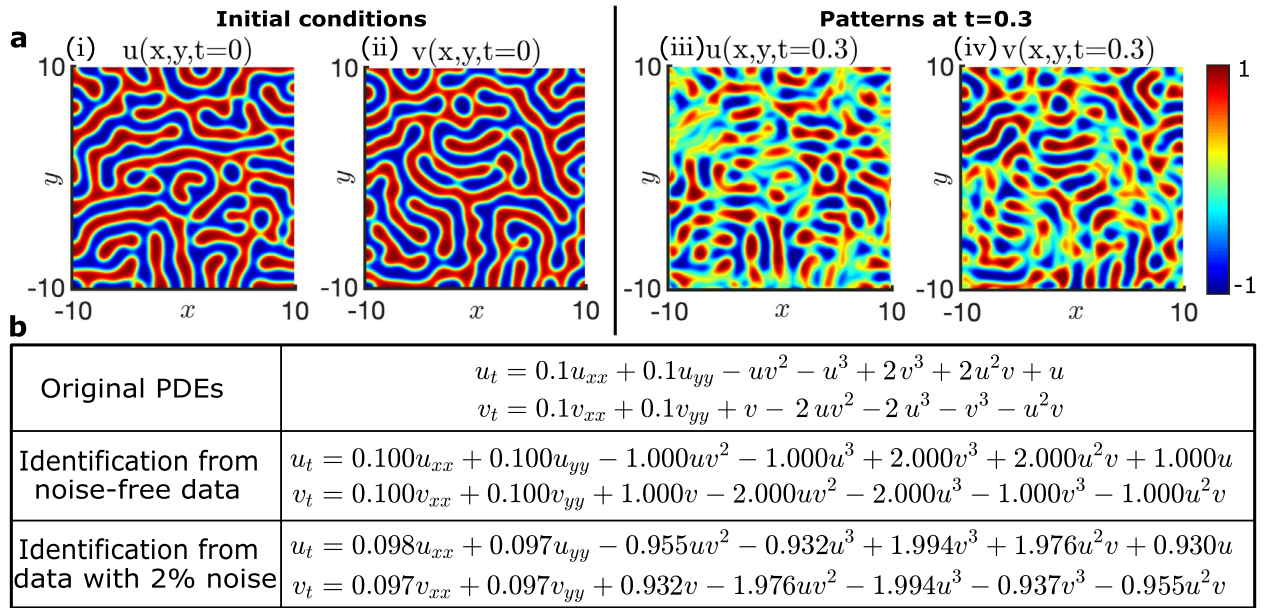
a

Initial conditions

(i) u(x,y,t=0)

(ii) v(x,y,t=0)

Patterns at t=0.3

(iii) u(x,y,t=0.3)

(iv) v(x,y,t=0.3)

Original PDEs	$u_t = 0.1u_{xx} + 0.1u_{yy} - uv^2 - u^3 + 2v^3 + 2u^2v + u$ $v_t = 0.1v_{xx} + 0.1v_{yy} + v - 2uv^2 - 2u^3 - v^3 - u^2v$
Identification from noise-free data	$u_t = 0.100u_{xx} + 0.100u_{yy} - 1.000uv^2 - 1.000u^3 + 2.000v^3 + 2.000u^2v + 1.000u$ $v_t = 0.100v_{xx} + 0.100v_{yy} + 1.000v - 2.000uv^2 - 2.000u^3 - 1.000v^3 - 1.000u^2v$
Identification from data with 2% noise	$u_t = 0.098u_{xx} + 0.097u_{yy} - 0.955uv^2 - 0.932u^3 + 1.994v^3 + 1.976u^2v + 0.930u$ $v_t = 0.097v_{xx} + 0.097v_{yy} + 0.932v - 1.976uv^2 - 1.994u^3 - 0.937v^3 - 0.955u^2v$

Figure 5. Demonstration of data-driven discovery of PDEs with ATSBL using a reaction–diffusion system. (**a-i,ii**) Snapshots of the initial conditions for the variables *u* and *v*, respectively. (**a-iii,iv**) *u* and *v* at time $t = 0.3$. (**b**) The table shows the original PDEs for the reaction–diffusion system and the identified PDEs for noise-free data and data with 2% Gaussian noise. Inference is conducted with a library containing 35 terms. The numerical calculations are done with a time step $\Delta t = 0.0034$ in the time interval $t = [0, 0.6]$. The space domain has size 20×20 and is covered with a 256×256 grid with periodic boundary conditions.

Function: AISO()
$\mathbf{c}^0 = \mathbf{0}$; %Initialize control force to zero
N = 10; %Fix number of iteration steps
i = 0; %Initialize iterator to zero
while $i < N$ **do**
    i= i+1;
    %For all degrees of freedom $\ell, \ell' \in \{1 \ldots M\}$
    %Add control force and generate data
    $\frac{dX_\ell(t)}{dt} = g_\ell(\mathbf{X}(t),t) + c_\ell^{i-1} + \sum_{\ell'} h_{\ell,\ell'}(\mathbf{X}(t),t)dW_{\ell'}(t)$
    %Subtract control force and collect data
    $\frac{dX_\ell(t)}{dt} \leftarrow \frac{dX_\ell(t)}{dt} - c_\ell^{i-1}$
    %Concatenate data
    $\Theta \leftarrow (\Theta, \Theta(X_\ell(t))$
    $\mathbf{g} \leftarrow (\mathbf{g}, \frac{dX_\ell(t)}{dt})$
    %Estimate libary coefficients with ATSBL
    $\mathbf{w} = \text{ATSBL}(\Theta, \mathbf{g}, d_{\text{tol}}, n_{\text{iters}})$
end
return **w**

Algorithm 2: Pseudocode for identification of SDEs with automatic iterative sampling optimization (AISO)

Identification of ordinary and partial differential equations. It is next shown that the sparse inference scheme based on Laplace priors that is implemented with ATSBL can also be used for data-driven discovery of ordinary and partial differential equations. The identification of ODEs from trajectory data is demonstrated with a Lorenz system, which is a paradigm for chaotic behavior[41]. The Lorenz equations are given by

$$\frac{\partial x}{\partial t} = a(y - x), \tag{30a}$$

$$\frac{\partial y}{\partial t} = x(b - z) - y, \tag{30b}$$

$$\frac{\partial z}{\partial t} = xy - cz, \tag{30c}$$

where the parameters are fixed as $a = 10$, $b = 28$, and $c = 3/8$. We numerically integrate these equations to obtain a trajectory as shown in Fig. 4a. The chaotic system involves two attractors. For data-driven system identification, we utilize three identical libraries Θ for each of the variables, x, y and z. Θ is constructed from the simulated trajectory and includes 56 terms containing up to fourth powers of all variables. Time-derivatives are calculated using fourth-order central-difference approximation. The general ODEs constructed from the library as in Eq. (4) are represented by three linear equation systems. The estimated equations resulting from an application of the inference procedure to noise-free data have small errors that are in magnitude comparable to the time step $\Delta t = 2 \times 10^{-4}$, see Fig. 4b. The same inference procedure is then repeated for a trajectory with additive Gaussian noise. The standard deviation of the noise in each coordinate is chosen to be 2% of the standard deviation of the noise-free data in the same coordinate. For this case, the ODEs identified with ATSBL still contain all the correct terms and the errors in the inferred system parameters are in the percent range, see Fig. 4b.

Finally, we demonstrate data-driven discovery of PDEs with ATSBL. Reaction–diffusion equations have attracted interest as prototypic models for pattern formation in biochemical systems, where constituents are locally transformed into each other through chemical reactions and transported in space by diffusion. Here, we consider the popular $\lambda - \omega$ system, given by

$$\frac{\partial u}{\partial t} = D_u \nabla^2 u + \lambda(A)u - \omega(A)v,$$
$$\frac{\partial v}{\partial t} = D_v \nabla^2 v + \omega(A)u + \lambda(A)v,$$
$$A = u^2 + v^2, \quad \omega = -\beta A^2, \quad \lambda = 1 - A^2,$$

where β is equal to 2. A two-dimensional, planar, rectangular area with periodic boundary conditions is considered. The initial values of u and v are shown in Fig. 5a-i,ii. The reaction–diffusion equations are solved numerically by using a spectral method. Snapshots of u and v are shown in Fig. 5a-iii,iv. For inference of the governing PDEs, a library matrix Θ is constructed containing 35 terms each for the time derivatives of u and v. Then, using ATSBL, the reaction–diffusion equations are inferred from the simulated data, as illustrated in Fig. 5b. For noise-free data, the identified equations deviate from the original equations only at the fourth decimal place and this error is due to discretization. However, if u and v are corrupted with additive noise, identification of the correct PDEs becomes challenging[4]. In Ref.[4], it has therefore been suggested to include a denoising step prior to the inference step. Accordingly, we employ a curvelet denoising method[42], which permits reconstruction of the reaction–diffusion equations from data with 2% noise with ATSBL, as illustrated in Fig. 5b.

Summary and outlook

Data-driven, automatic discovery of governing equations has become a viable tool for studying complex systems if first-principle derivations are intractable, e.g., for biological systems or epidemiological data. The aim is here generally to construct an analytical model that characterizes the observed dynamics and extends to parameter- and phase space regions that are hard to access experimentally.

Our main contribution is an inference method that makes use of Laplacian prior distributions in a Bayesian framework to find a minimal set of governing equations without the need for user input. We establish the validity of this approach and compare it to other methods. Regarding data-driven discovery of Langevin-type SDEs, we show that the proposed sparse method converges faster than other methods based on ridge regression. Maximum likelihood methods for the estimation of parameters in SDEs are not considered here, see Ref.[8] for an introduction to those methods. For the studied Langevin SDEs, we find that a binning of the trajectory data for inference of the drift and diffusion coefficients is only advantageous if the phase space is sampled sparsely. In that case, the error of the inference procedure can vary strongly in phase space since the relative uncertainties of the probabilities vary. A filtering procedure consisting of the exclusion of data with high uncertainty results in a significantly improved inference accuracy.

Next, we investigate how active-learning procedures can be useful when a direct inference of the global dynamics is difficult because of a trapping of trajectories in local energetic minima. This problem can be solved with well-established umbrella-sampling methods where quadratic bias potentials are employed to reduce the energetic barriers in the original potential landscape[18,43]. However, an appropriate parameterization of such bias potentials can be challenging. For example, if the bias potentials are intended to smoothen an unknown, rough energy landscape. Instead, we employ the methods for data-driven identification of governing equations to calculate time-dependent external perturbations that force the trajectory to explore the full phase space. The main feature of our method is that the parameters that determine the applied perturbations correspond to the parameters that define the energy landscape. The combination of iterative inference with system perturbations can significantly improve the speed and accuracy of the overall inference procedure. We therefore hope that the suggested active learning procedure will extend the applicability of data-driven methods, in particular in the context of computer simulations.

A central challenge related to the improvement of the library-based methodology for identification of analytical models is to find automated approaches for tailoring the employed function space to the problem at hand.

Recent methodological advances suggest that a possible solution is the integration of physical constraints, such as symmetries, conservation laws, or even thermodynamics, into a generic framework for statistical learning of governing equations[44]. Data-driven identification of analytical models thus has the potential to become a popular tool for closing the gap between non-parametric, empirical modeling and first-principles-based modeling in the coming years.

Data availability
Source code and data can be obtained from the corresponding author upon request.

References
 1. Schmidt, M. & Lipson, H. Distilling free-form natural laws from experimental data. *Science* **324**, 81 (2009).
 2. Bongard, J. & Lipson, H. Automated reverse engineering of nonlinear dynamical systems. *Proc. Natl. Acad. Sci. U.S.A.* **104**, 9943 (2007).
 3. Brunton, S. L., Proctor, J. L. & Kutz, J. N. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proc. Natl. Acad. Sci. U.S.A.* **113**, 3932 (2016).
 4. Rudy, S. H., Brunton, S. L., Proctor, J. L. & Kutz, J. N. Data-driven discovery of partial differential equations. *Sci. Adv.* **3**, e1602614 (2017).
 5. Boninsegna, L., Nüske, F. & Clementi, C. Sparse learning of stochastic dynamical equations. *J. Chem. Phys.* **148**, 241723 (2018).
 6. Raissi, M., Yazdani, A. & Karniadakis, G. E. Hidden fluid mechanics: Learning velocity and pressure fields from flow visualizations. *Science* **367**, 1026 (2020).
 7. Zhang, S. & Lin, G. Robust data-driven discovery of governing physical laws with error bars. *Proc. Math. Phys. Eng.* **474**, 20180305 (2018).
 8. Bishwal, J. P. *Parameter Estimation in Stochastic Differential Equations* (Springer, 2007).
 9. Friedrich, R., Peinke, J., Sahimi, M. & Tabar, M. R. R. Approaching complexity by stochastic methods: From biological systems to turbulence. *Phys. Rep.* **506**, 87 (2011).
 10. Stephens, G. J., De Mesquita, M. B., Ryu, W. S. & Bialek, W. Emergence of long timescales and stereotyped behaviors in Caenorhabditis elegans. *Proc. Natl. Acad. Sci. U.S.A.* **108**, 7286 (2011).
 11. Sarfati, R., Błwzdziewicz, J. & Dufresne, E. R. Maximum likelihood estimations of force and mobility from single short Brownian trajectories. *Soft Matter* **13**, 2174 (2017).
 12. Pérez García, L., Donlucas Pérez, J., Volpe, G., Arzola, A. V. & Volpe, G. High-performance reconstruction of microscopic force fields from Brownian trajectories. *Nat. Commun.* **9**, 1 (2018).
 13. Baldovin, M., Puglisi, A. & Vulpiani, A. Langevin equations from experimental data: The case of rotational diffusion in granular media. *PLoS ONE* **14**, e0212135 (2019).
 14. Frishman, A. & Ronceray, P. Learning force fields from stochastic trajectories. *Phys. Rev. X* **10**, 021009 (2020).
 15. Ferretti, F., Chardès, V., Mora, T., Walczak, A. M. & Giardina, I. Building general Langevin models from discrete datasets. *Phys. Rev. X* **10**, 031018 (2020).
 16. Brückner, D. B., Ronceray, P. & Broedersz, C. P. Inferring the dynamics of underdamped stochastic systems. *Phys. Rev. Lett.* **125**, 058103 (2020).
 17. Brückner, D. B. *et al.* Learning the dynamics of cell–cell interactions in confined cell migration. *Proc. Natl. Acad. Sci. U.S.A.* **118**, e2016602118 (2021).
 18. Torrie, G. M. & Valleau, J. P. Nonphysical sampling distributions in Monte Carlo free-energy estimation: Umbrella sampling. *J. Comput. Phys.* **23**, 187 (1977).
 19. Valsson, O. & Parrinello, M. Variational approach to enhanced sampling and free energy calculations. *Phys. Rev. Lett.* **113**, 090601 (2014).
 20. Dama, J. F., Parrinello, M. & Voth, G. A. Well-tempered metadynamics converges asymptotically. *Phys. Rev. Lett.* **112**, 240602 (2014).
 21. Invernizzi, M., Piaggi, P. M. & Parrinello, M. Unified approach to enhanced sampling. *Phys. Rev. X* **10**, 041034 (2020).
 22. Besold, G., Risbo, J. & Mouritsen, O. G. Efficient Monte Carlo sampling by direct flattening of free energy barriers. *Comput. Mater. Sci.* **15**, 311 (1999).
 23. Goedecker, S. Minima hopping: An efficient search method for the global minimum of the potential energy surface of complex molecular systems. *J. Chem. Phys.* **120**, 9911 (2004).
 24. Blaak, R., Auer, S., Frenkel, D. & Löwen, H. Crystal nucleation of colloidal suspensions under shear. *Phys. Rev. Lett.* **93**, 068303 (2004).
 25. Klymko, K., Geissler, P. L., Garrahan, J. P. & Whitelam, S. Rare behavior of growth processes via umbrella sampling of trajectories. *Phys. Rev. E* **97**, 032123 (2018).
 26. Risken, H. Fokker–Planck equation. In *The Fokker–Planck Equation* 63–95 (Springer, 1996).
 27. Kleinhans, D., Friedrich, R., Wächter, M. & Peinke, J. Markov properties in presence of measurement noise. *Phys. Rev. E* **76**, 041109 (2007).
 28. Ragwitz, M. & Kantz, H. Indispensable finite time corrections for Fokker–Planck equations from time series data. *Phys. Rev. Lett.* **87**, 254501 (2001).
 29. Friedrich, R., Renner, C., Siefert, M. & Peinke, J. Comment on "indispensable finite time corrections for Fokker–Planck equations from time series data". *Phys. Rev. Lett.* **89**, 149401 (2002).
 30. Gottschall, J. & Peinke, J. On the definition and handling of different drift and diffusion estimates. *New J. Phys.* **10**, 083034 (2008).
 31. Honisch, C. & Friedrich, R. Estimation of Kramers–Moyal coefficients at low sampling rates. *Phys. Rev. E* **83**, 066701 (2011).
 32. Rydin Gorjão, L., Witthaut, D., Lehnertz, K. & Lind, P. G. Arbitrary-order finite-time corrections for the Kramers–Moyal operator. *Entropy* **23**, 517 (2021).
 33. Rinn, P., Lind, P., Wächter, M. & Peinke, J. The Langevin approach: An R package for modeling Markov processes. *J. Open Res. Softw.* **4** (2016).
 34. Gradišek, J., Siegert, S., Friedrich, R. & Grabec, I. Analysis of time series from stochastic processes. *Phys. Rev. E* **62**, 3146 (2000).
 35. El-Sayed, M. A. A new algorithm based entropic threshold for edge detection in images. *Int. J. Comput. Sci.* **8**, 71 (2011).
 36. Hamza, A. B. Nonextensive information-theoretic measure for image edge detection. *J. Electron. Imaging* **15**, 013011 (2006).
 37. Babacan, S. D., Molina, R. & Katsaggelos, A. K. Bayesian compressive sensing using Laplace priors. *IEEE Trans. Image Process.* **19**, 53 (2009).
 38. Tipping, M. E. Sparse Bayesian learning and the relevance vector machine. *J. Mach. Learn. Res.* **1**, 211 (2001).
 39. Ji, S., Xue, Y. & Carin, L. Bayesian compressive sensing. *IEEE Trans. Signal* **56**, 2346 (2008).

16

40. Figueiredo, M. A. Adaptive sparseness for supervised learning. *IEEE Trans. Pattern Anal. Mach. Intell.* **25**, 1150 (2003).
41. Lorenz, E. N. Deterministic nonperiodic flow. *J. Atmos. Sci.* **20**, 130 (1963).
42. Peyré, G. The numerical tours of signal processing-advanced computational signal and image processing. *IEEE Comput. Sci. Eng.* **13**, 94 (2011).
43. Bussi, G. & Laio, A. Using metadynamics to explore complex free-energy landscapes. *Nat. Rev. Phys.* **2**, 200 (2020).
44. Karniadakis, G. E. *et al.* Physics-informed machine learning. *Nat. Rev. Phys.* **3**, 422 (2021).

Author contributions

All authors designed the study, performed the research and wrote the manuscript together.

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to B.S.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.